Finite Difference Poisson–Boltzmann Electrostatic Calculations: Increased Accuracy Achieved by Harmonic Dielectric Smoothing and Charge Antialiasing

ROBERT E. BRUCCOLERI,* JIRI NOVOTNY, and MALCOLM E. DAVIS

Bristol–Myers Squibb Pharmaceutical Research Institute, Department of Macromolecular Modeling, P.O. Box 4000, Princeton, New Jersey 08543-4000

KIM A. SHARP

Johnson Research Foundation, Department of Biochemistry and Biophysics, University of Pennsylvania, 37th & Hamilton Walk, Philadelphia, Pennsylvania 19104-6059

Received 1 February 1996; accepted 9 May 1996

ABSTRACT

A common problem in the calculation of electrostatic potentials with the Poisson–Boltzmann equation using finite difference methods is the effect of molecular position relative to the grid. Previously a uniform charging method was shown to reduce the grid dependence substantially over the point charge model used in commercially available codes. In this article we demonstrate that smoothing the charge and dielectric values on the grid can improve the grid independence, as measured by the spread of calculated values, by another order of magnitude. Calculations of Born ion solvation energies, small molecule solvation energies, the electrostatic field of superoxide dismutase, and protein–protein binding energies are used to demonstrate that this method yields the same results as the point charge model while reducing the positional errors by several orders of magnitude.

* Author to whom correspondence should be addressed.
E-mail: bruc@bms.com

Journal of Computational Chemistry, Vol. 18, No. 2, 268–276 (1997)

CCC 0192-8651 / 97 / 020268-09

Introduction

The representation of solvent in molecular simulations has always been a difficult problem. All such representations can be divided into two large categories: explicit models, where the solvent is represented by a large number of individual solvent molecules, and implicit models, where the effects of solvent are incorporated by treating it as a continuum dielectric.

One of the most successful implicit models for the treatment of electrostatic effects is the Poisson–Boltzmann (PB) equation.[1] The equation can be solved over a volume enclosing a molecule of interest and a region of surrounding solvent using finite difference (FD) methods[2–4] appropriate for the solution of boundary value problems (the FDPB method). In these methods, a grid is laid down over the volume, and the electrostatic properties of the system are computed over discrete points in space. In a previous article[5] it was demonstrated that treating each atom as a uniformly charged sphere instead of a point charge markedly improved the dependence of the electrostatic energy on the position of the molecule relative to the grid.

To further improve on the grid positioning independence, it is necessary to reduce the errors associated with the discrete representation required by FD methods. Similar errors stemming from discrete representations arise in computer graphics, where continuous 2-dimensional objects, such as lines, curves, polygons, and images, must be mapped onto a raster display. By analogy to techniques in computer graphics, we developed two significant improvements in the construction of the grid. First, we implemented an antialiasing approach for the charging of the grid that smooths the edge of each atomic sphere. Second, we implemented a smoothing scheme for the dielectric values that better approximates the transition from the molecular interior to the solvent. These two boundary smoothing algorithms significantly lessen the effect of moving the molecular system on the grid and make possible the direct comparison of electrostatic energies for different conformers of the same molecule. In addition, we also modified the GEPOL93 molecular surface algorithm[6] to improve its performance and incorporated it into the program, CONGEN[7,8] for use in the solution of the PB equation.

Methods

The CONGEN molecular modeling program, which incorporates an FDPB solver, was used to obtain solutions to the PB equation. Three new methods were introduced into the PB solver: charge antialiasing, dielectric smoothing, and the use of the GEPOL93 algorithm for generating the molecular surface. In the FDPB method, the molecule and a region of the surrounding solvent is mapped onto a rectangular grid composed of cubic cells. The spacing of the grid is determined by a balance between achieving a fine representation of the molecule and keeping a sufficient layer of solvent between the molecule and the grid boundary so that the boundary potentials may be determined accurately. After scaling and positioning the molecule on the grid, values for the charge, dielectric constant, and ionic strength (if salt is present in the solvent) must be assigned the appropriate values everywhere on the grid. The discretization of space implicit in the FD method effectively subdivides the space in and around the molecular system into small cubes. Each cube has an associated charge density that is constant across the cube, and the face of each cube has an associated dielectric constant that is shared with the adjoining cube.

CHARGE ANTIALIASING

Previously the setting of the charge density for each cube was an all or nothing decision, depending on whether that cube lay within an atom that was being charged.[5] However, by analogy to the solution of the antialiasing problem in the drawing of lines on a raster graphics device (ref. 9, pp. 132–140), one can improve upon the charge representation by setting the charge density in proportion to the fraction of the cube that is occupied by each atom. Grid points just outside the van der Waals surface are also checked.

The algorithm for charge antialiasing is straightforward. For each atom, CONGEN loops over each grid point within a distance equal to that atom's van der Waals radius plus one grid spacing. The cube around each grid point is evenly subdivided in all three Cartesian directions, and the program counts the number of subcubes whose centers lie within the van der Waals radius of the atom. The charge of the atom is then divided by the total subcube count to get the subcube charge density.

The loop over the grid points is repeated, and the charge assigned to each grid point is set to the subcube charge density multiplied by the number of subcubes within the van der Waals radius. Figure 1 illustrates the process in two dimensions. Tests on the Born ion system showed that convergence of results is obtained with a subdivision of five in each dimension, i.e., 125 subcubes per grid point.

DIELECTRIC SMOOTHING

The discretization of space also affects the modeled shape of the dielectric. Without smoothing, the dielectric is modeled as a set of juxtaposed cubes of discrete dielectric constant values. It is not possible to use an antialiasing method because in the FDPB method, the protein/solvent dielectric boundary is taken to be the molecular surface.[10] The molecular surface is composed partly of the van der Waals surface of the atoms, partly from the reentrant surface from the generating probe that is usually a water molecule sized sphere.[11] Thus, the dielectric constant of a point in space is not determined solely by whether any atom's van der Waals surface overlays that point. The dielectric boundaries can be smoothed only after all the atoms in the system have been mapped onto the grid. Therefore, a volume filtering approach is used to smooth the dielectric constant values on the grid.

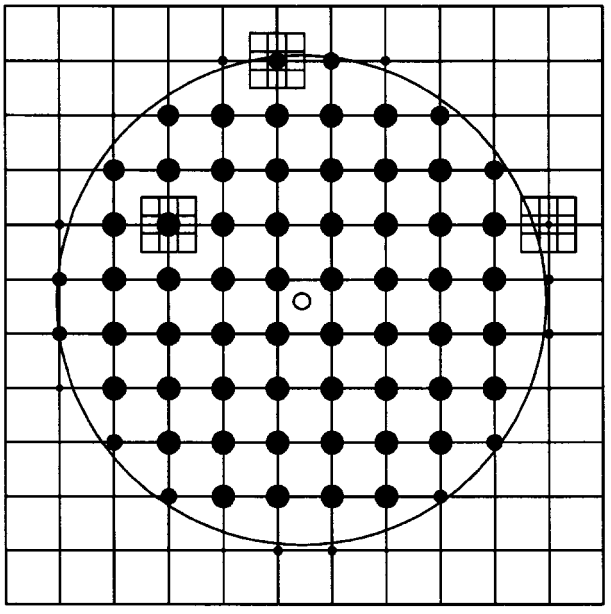

FIGURE 1. A 2-D representation of the charge antialiasing. The small open circle in the center represents the atomic center, and the large circle is the van der Waals surface. At each point, the space is further subdivided into smaller cubes (in this example, the subdivision is done three times in the 2 dimensions of the drawing). The number of subcube centers covered by the van der Waals surface is then tallied (in this example, the number is 573). The charge on each point in space is given in proportion to the number of subcubes covered by the atom. In this illustration, we show subcubes for three different points. The leftmost point has all nine subcubes covered, and it will get a charge of 9 / 573 of the atom. The middle point at the top edge will receive a charge of 6 / 573 because six subcubes are covered, and the rightmost point will get a charge of 2 / 573 of the atom. The algorithm must check points outside the van der Waals surface to a distance of one grid spacing to account for all subcubes that can be covered by the atom.

The filtering is performed by first copying the dielectric grid and then looping over each point in the grid, calculating an average of the dielectric over a suitable number of neighboring grid points, and storing these averages in the new grid. Several filtering options were explored. Averaging can be performed either arithmetically using the expression, $(1/n)\Sigma_i \varepsilon_i$, or harmonically using the expression, $n/(\Sigma_i 1/\varepsilon_i)$. In these expressions, i varies over the spatial nearby points. Testing shows that the harmonic average is the correct approach. This method is conceptually related to the dielectric boundary smoothing method of Davis and McCammon[12].

Another filtering option was the selection of neighboring points. In the FD method, the points assigned a dielectric value lie at the midpoint of the cubic faces. In other words, they lie on the midpoints of the lines joining the charge bearing grid points. On a cubic grid, each dielectric grid point has eight nearest neighbors lying $1/\sqrt{2}$ grid units away, and six next nearest neighbors lying 1 grid unit away as shown in Figure 2.

We investigated averaging performed over the nearest nine or 15 grid points or over a nearby spherical volume whose radial dimension can be arbitrarily specified in angstroms. The purpose of grid point averaging is smoothing where the localization is based on the grid spacing. On the other hand, volume averaging is based on the physical dimension of the system and performs averaging over all grid points within a given distance to each grid point.

Here, the grid averaging was found to be preferable because it operates on the same dimension as the grid; therefore, it works to smooth just

the atomic surfaces at the resolution of the grid. In addition, the volume based averaging can be very slow because of the large number of points that can be summed.

Finally, the dielectric values can be either constant weighted or Gaussian weighted by distance; i.e., each permittivity value can be scaled by $\exp(-(\alpha r)^2)$, with α being a user selected parameter, and r being the distance of the neighboring point to the point being averaged. We found no improvement in the calculations using small Gaussian weighting parameters ($\alpha = 0.6$), so constant weighting was used throughout the calculations.

MOLECULAR SURFACE

The GEPOL93 molecular surface algorithm[6] was incorporated into CONGEN with modifications to improve its performance on large molecular systems. The molecular surface is found as an envelope of fictitious spheres interpolated between pairs of atomic spheres. This method is well suited for delineating the dielectric boundary of a molecule because only simple distance tests are required from each grid point to the center of the atomic or fictitious molecular surface spheres.

The GEPOL93 source code was modified to use the space grid in CONGEN to identify the neighbors of atoms or points in space. The original algorithm used triply nested loops over all the atoms or spheres in the system, which gives $O(n^3)$ asymptotic time complexity. The use of the space grid changes the asymptotic time complexity to $O(n^2)$ but with some performance penalty for small molecules.

The additional computer time necessary to execute the above three algorithms is comparable to the CPU time required to solve for the electrostatic potential.

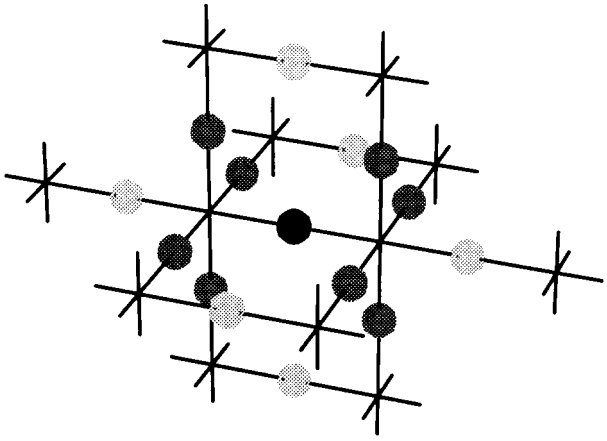

FIGURE 2. An illustration of the dielectric grid points that are averaged. The intersections of the lines represent grid points where charges are stored. Dielectric values are stored at points in space halfway between each grid point along each of the orthogonal coordinate axes. In this figure, the dielectric point about which the average is to be computed is shown with a black circle. The eight closest dielectric grid points used in 9-point smoothing are shown in dark gray. These are all located at a distance of $h/\sqrt{2}$ from the central point, where h is the grid spacing. The next six nearest dielectric grid points are shown in light gray and are at a distance of h from the central point. These six points contribute to the 15-point smoothing.

Results and Discussion

As before we used ion solvation, molecular solvation, and the potential around superoxide dismutase to test the new methods.[5] In addition, we illustrate the effect of these improvements on the calculation of electrostatic binding energy in a large protein–protein complex, the binding of the Fv domain of HyHEL-10 with lysozyme.[13]

ION SOLVATION

The Born ion solvation model[14] is the simplest test of our methods. A single ion of unit charge and 2 Å radius is placed in the grid; the potentials on the grid boundary are set using the Debye–Huckel point potential approximation described previously,[5, 10] because this rule specifies the correct boundary values as determined by continuum electrostatic theory. The potential is calculated using external dielectrics of 1 and 78, and the difference in electrostatic energy, $1/2\int\phi\rho\, dV$, calculated for each external dielectric, gives the solvation energy. The results are compared against the exact solution for this model

$$E = \frac{k_e q^2}{2r}\left(\frac{1}{\varepsilon_s} - \frac{1}{\varepsilon_0}\right), \qquad (1)$$

where E is the solvation energy, k_e is the electrostatic force constant, q is the ionic charge, r is the atomic radius, ε_s is the solvent dielectric, and ε_0 is the vacuum dielectric defined as 1.

Figure 3a illustrates the effect of charge antialiasing and dielectric smoothing on the calcula-

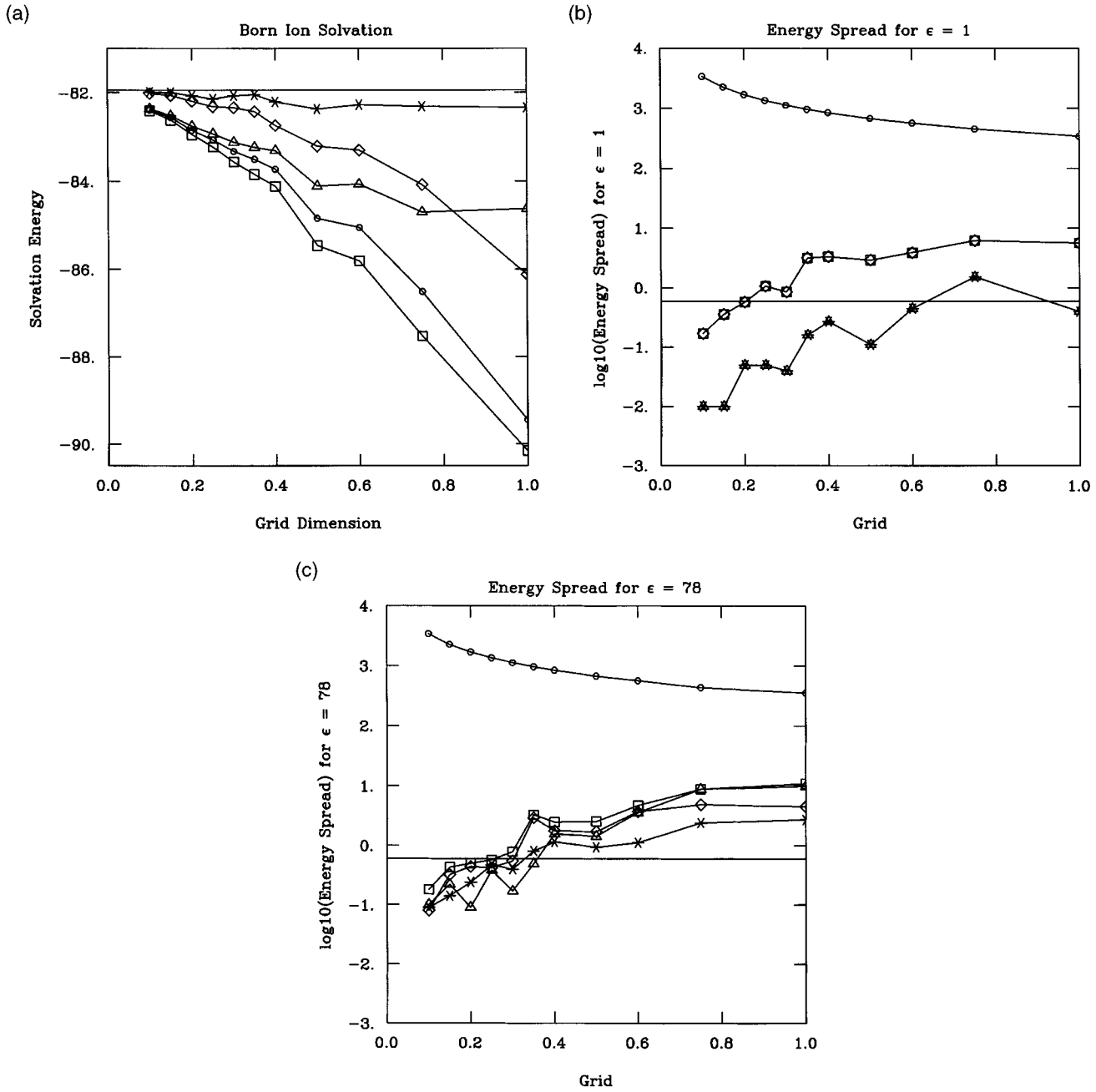

FIGURE 3. Calculation of Born ion solvation energy as a function of grid spacing and setup. (a) The solvation energy calculation. Each point represents the average of 10 different Poisson–Boltzmann calculations using the following locations of the ion on the grid: placed on a grid point; centered in the space between grid points; positioned on the *x*, *y*, and *z* lattice lines; and five random positions within the grid. The horizontal bar shows the theoretical Born ion solvation energy for a sphere of radius 2 Å in a media of dielectric, $\varepsilon = 78$. (b) The spread in energies for the calculation of the energy of the ion in a vacuum. The spread in energy is the difference between the largest and smallest self-energy across the 10 different grid positions. Note that the energy spread is on a *logarithmic* scale. Also, dielectric smoothing has no effect because the entire system is at a dielectric, $\varepsilon = 1$. (c) The spread in energies for the calculation of the energy in a medium of dielectric, $\varepsilon = 78$. In all panels, the symbols have the following meaning: (○) trilinear interpolation; (□) uniform charging; (△) antialiased charging, subdivision of five; (◇) uniform charging and dielectric smoothed, grid based 15 point, uniform weighting; (∗) antialiased charging and dielectric smoothing as above.

tion of the ion solvation energy as a function of the grid spacing. The combination of charge antialiasing and dielectric smoothing nearly eliminates the error in the solvation energy calculation in the range of grid spacings of 0.1–1.0 Å. Figures 3b and c show that electrostatic energy calculations converge with decreasing grid spacing when any of these uniformly charged sphere approximations are used, but that the point charge model (trilinear interpolation) diverges as the grid spacing shrinks. For the self-energy of a sphere in a vacuum, the charge antialiasing method gives an order of magnitude better convergence, whereas all of the uniformly charged sphere approximations give similar convergence for the self-energy of a sphere in solvent.

MOLECULAR SOLVATION

The next test of the improved grid initializations is the calculation of the solvation energy of nine different molecules. In the previous study a similar test was performed, except that the solvation of a small protein, bovine pancreatic trypsin inhibitor (BPTI), was not included.[5] Table I shows a comparison of the solvation energy calculations for three different charge assignment methods: trilinear interpolation, uniform charging (as in the previous article[5]), and dielectric smoothing with charge antialiasing. The results of the solvation energy calculations are all similar, but the variations are substantially reduced when the charge antialiasing and dielectric smoothing are used. In

TABLE I.
Grid Independence of Solvation Energy Calculations.

Molecule	Trilinear[a] E_{solv}[d]	Trilinear[a] Range[e] $E_{\varepsilon=78}$	Uniform[b] E_{solv}[d]	Uniform[b] Range[e] $E_{\varepsilon=78}$	Antialiased and Smoothed[c] E_{solv}[d]	Antialiased and Smoothed[c] Range[e] $E_{\varepsilon=78}$
Methanol	−8.32 ± 0.20	308.2	−8.46 ± 0.34	2.14	−7.82 ± 0.03	0.22
Ethanol	−8.13 ± 0.21	190.6	−8.35 ± 0.55	0.90	−7.55 ± 0.06	0.31
2-propanol	−8.00 ± 0.29	335.6	−8.05 ± 0.63	1.84	−7.41 ± 0.09	0.45
Acetone	−4.07 ± 0.03	126.5	−4.11 ± 0.04	0.22	−3.88 ± 0.01	0.03
CH_3-acetate	−3.38 ± 0.02	168.0	−3.45 ± 0.04	0.17	−3.17 ± 0.01	0.03
Acetic acid	−77.36 ± 0.11	436.3	−77.53 ± 0.20	1.15	−76.00 ± 0.04	0.14
Acetamide	−13.22 ± 0.30	553.3	−13.21 ± 0.46	2.74	−12.72 ± 0.06	0.48
C7eq Ala[f]	−14.19 ± 0.21	329.9	−14.35 ± 0.39	2.10	−13.36 ± 0.08	0.53
C5 Ala[f]	−17.29 ± 0.31	430.6	−17.62 ± 0.46	2.58	−16.36 ± 0.09	0.32
α_R Ala[f]	−22.13 ± 0.20	407.3	−22.13 ± 0.44	1.13	−21.13 ± 0.08	0.41
BPTI[g]	−1272.73 ± 0.82	3291.4	−1277.18 ± 1.30	14.68	−1239.80 ± 0.31	1.15

Solvation energy calculations of these molecules were performed over 10 different random grid placements using three different grid charging schemes. All energies are given in units of kcal / mol. The charges and geometric parameters for the atoms in the small molecules are taken from the Optimized Parameters for Liquid Simulations (OPLS) parameters,[17] except that the radius of all charged hydrogens is 0.8 Å. In OPLS, the hydrogen radii are set to zero to improve the calculation of hydrogen bond energy, but such a radius is not appropriate for a finite difference calculation. The grid spacing is 0.3 Å, and the size of the cubic box is 15 Å. The ionic strength is set to zero, and the interior dielectric constant is set to 1. The solvent is presumed to begin at the van der Waals radius of the atoms. Coordinates for the small molecules were generated as described previously.[5] The CPU time for the 10 small molecule calculations (including all the grid placements) was 54 min on a 150-MHz SGI R4400 system. The CPU time for the BPTI calculation was 29 h on the same processor.

[a] Trilinear interpolation was used for charging the grid.

[b] Uniform sphere charging was used for charging the grid.

[c] Antialiased charging and dielectric smoothing were used.

[d] The solvation energy averaged over the 10 different grid placements. Values are mean ± the standard deviation from the average solvation energy.

[e] The range is the difference between the maximum and minimum electrostatic energy for the molecule in the solvent. The ranges for the electrostatic energy in a vacuum are similar.

[f] These examples are different conformations of the alanine dipeptide. Original conformations are defined in Scarsdale et al.[18]

[g] Calculations with bovine pancreatic trypsin inhibitor were made using the molecular surface to define the interior. In addition, the four water molecules in the interior of the protein were included in the low dielectric interior. Also, these calculations were performed using the AMBER potential[19] and the size of the box is 45 Å.

addition, the errors encountered as the molecules are shifted on the grid is likewise reduced. The case of BPTI is especially striking in spite of the large size of the system (594 atoms). The relative error associated with the charged antialiased and dielectric smoothed evaluation of its solvation energy does not exceed 2.5×10^{-4}. Thus, these new methods allow the direct comparison of electrostatic energies calculated at different placements and with different conformations.

ELECTROSTATIC POTENTIAL OF SUPEROXIDE DISMUTASE

The third test of the improved initialization algorithm was the calculation of electrostatic potential of a large molecule, superoxide dismutase.[15] The goal of this test was to verify that the new method could reproduce previous calculations of the potential[16] that used trilinear interpolation.

The calculation of the electrostatic potential of superoxide dismutase was performed at one grid spacing, 0.6 Å, and using either the trilinear charging models or the combination of charge antialiasing and dielectric smoothing. The external potentials are visually superimposable as found previously.[5] Table II shows a summary of electrostatic potential differences outside superoxide dismutase. As can be seen in this table, the differences in the potential as a function of the charging model are very small.

TABLE II.
Potential Differences for Superoxide Dismutase.

Range (kT/e)	Points[a]	Fraction[b]
$-0.7 : -0.5$	2	9.40949×10^{-7}
$-0.5 : -0.3$	86	4.04608×10^{-5}
$-0.3 : -0.1$	3,576	1.68242×10^{-3}
$-0.1 : 0.1$	2,116,085	0.995564
$0.1 : 0.3$	4,966	2.33638×10^{-3}
$0.3 : 0.5$	530	2.49352×10^{-4}
$0.5 : 0.7$	174	8.18626×10^{-5}
$0.7 : 0.9$	64	3.01104×10^{-5}
$0.9 : 1.1$	20	9.40949×10^{-6}
$1.1 : 1.3$	8	3.7638×10^{-6}
$1.3 : 1.5$	2	9.40949×10^{-7}

The electrostatic potential around superoxide dismutase was calculated using either trilinear interpolation or the combination of antialiased charging and dielectric smoothing. A grid spacing of 0.6 Å was used, and the dimensions were $167 \times 118 \times 123$ points. The molecular surface was used to define the interior of the molecule. The differences between these calculations were determined at all grid points outside the Stern layer (2 Å) of the protein. These differences were categorized into ranges of size 0.10 kT/e where T is 300°. The solvent dielectric for these calculations was 80, and the interior dielectric was 2. The water radius was 1.4 Å, and the ionic strength was 0.169 M, which corresponds to a Debye length of 7.5 Å. The boundary was set using the atomic Debye rule, and the points in the boundary are included in the above tabulation.

[a] This is the number of points in the grid outside the protein's Stern layer whose potential difference is within the given range.

[b] This is the fraction of points in the grid outside the molecule whose potential difference is within the given range. In this calculation, there are 2,125,513 such points.

ANTIBODY LYSOZYME ELECTROSTATIC BINDING ENERGY

The most ambitious test of the improved algorithms is the calculation of the electrostatic binding energy of the complex between the Fv domain of HyHEL-10 and hen egg white lysozyme.[13] Not only does the size of the system exceed 3300 heavy atoms and charged hydrogens, but the calculation calls for independent evaluation of electrostatic energies of the macromolecular complex and its separate parts and their subtraction. We performed this subtractive calculation 10 times while varying the positions of the molecules on the grid by successive 1° rotations about the vector (3, 2, 1).

Figure 4 shows the electrostatic binding energy as a function of the grid spacing. The variation in the calculated energy values is due to the rotation of the complex on the grid and is shown by the error bars. It is clear that the improved initialization algorithm results in good convergence and permits the calculation to be accurately performed with a much higher grid spacing than the other methods. Based on Figure 4, 0.8 Å can be used as the grid spacing for routine binding energy calculations, because convergence was achieved using the combination of charge antialiasing and dielectric smoothing.

CPU time and memory usage are substantially affected by the different grid sizes. A grid size of 0.3 Å requires 1803 megabytes (MB) of memory vs. 120 MB for the 0.8-Å grid size, i.e., a ratio of 15. The total CPU time requirement for one binding energy calculation is about 1 h on a single 150-MHz SGI R4400 system for a grid size of 0.8 Å, and about 24 h on the same processor for a grid size of 0.3 Å. The CPU time includes both grid setup and solving for the potential. For the 0.8 Å grid size, approximately half the time is used for setup, whereas for the 0.3-Å grid size, about 40% of the time is used for setup.

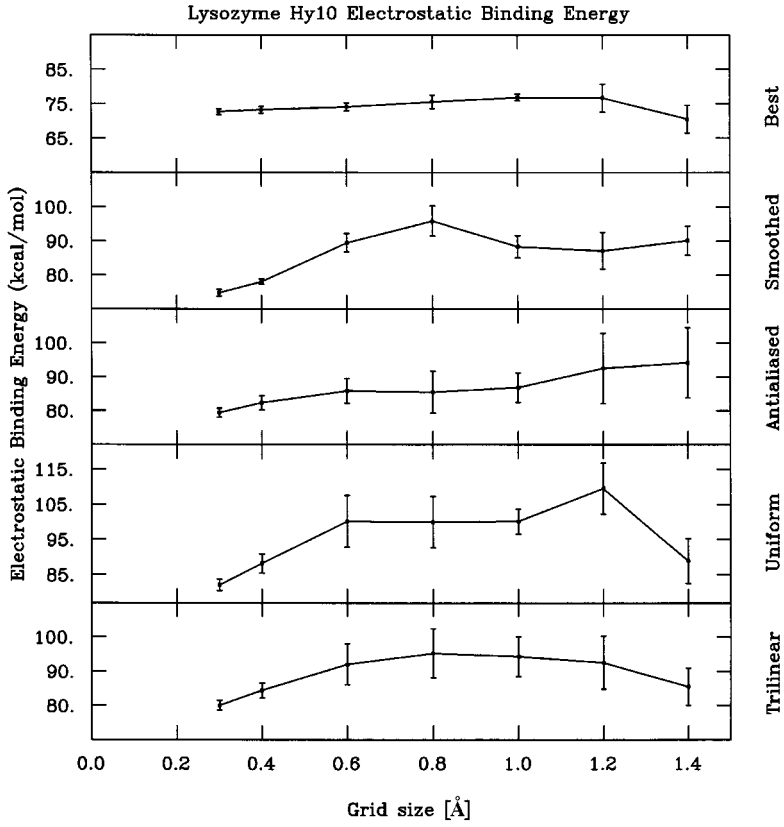

FIGURE 4. Grid dependence of lysozyme–HyHEL-10 binding energy calculations. Electrostatic binding energies for the HyHEL-10 lysozyme complex[13] were calculated over 10 different grid placements generated by rotating the system by 1° increments about the vector (3, 2, 1). The dots represent the average electrostatic binding energy, and the error bars are twice the standard deviation of the distribution of calculated electrostatic binding energies. The captions at the right of each panel have the following meaning: trilinear, trilinear interpolation; uniform, uniform charging; antialiased, antialiased charging, subdivision of 5; smoothed, uniform charging and dielectric smoothed, grid based 15 point, uniform weighting; best, antialiased charging and dielectric smoothing as above.

CONCLUSIONS

We described a significant improvement in the grid initialization algorithm in an FDPB calculation. These improvements are simple to implement, and they substantially improve upon the grid independence of the uniform charging algorithm described previously. In addition, they improve on the quality of the PB energy calculation so that very small grids are not required for accurate calculations. For electrostatic binding energies, a comfortable grid spacing (0.8 Å) suffices for acceptable accuracy (under 1 kcal/mol). This has a pronounced effect on the time and memory requirements of such calculations.

For information about obtaining the CONGEN program, please contact R. E. Bruccoleri.

Acknowledgments

We thank Alissa Rashkin and Xiaoxi Chen for helpful discussion. K. A. S. thanks the E. R. Johnson Foundation and the NSF (MCB95-06900) for financial support.

References

1. S. C. Harvey, *Proteins: Struct. Funct. Genet.*, **5**, 78 (1989).
2. J. Warwicker and H. C. Watson, *J. Mol. Biol.*, **157**, 671 (1982).
3. M. E. Davis and J. A. McCammon, *J. Comput. Chem.*, **10**, 386 (1989).
4. A. Nicholls and B. Honig, *J. Comput. Chem.*, **12**, 435 (1991).
5. R. E. Bruccoleri, *J. Comput. Chem.*, **14**, 1417 (1993).

6. J. L. Pascual−Ahuir, E. Silla, and I. Tuñon, *J. Comput. Chem.*, **15**, 1127 (1994).
7. R. E. Bruccoleri, *CONGEN v2.1 Manual*, Bristol−Myers Squibb Pharmaceutical Research Institute, Princeton, NJ, 1996.
8. R. E. Bruccoleri, *Mol. Simulation*, **10**, 151 (1993).
9. J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice*, 2nd ed., Addison−Wesley, Reading, MA, 1990.
10. M. K. Gilson, K. A. Sharp, and B. H. Honig, *J. Comput. Chem.*, **9**, 327 (1987).
11. B.-K. Lee and F. M. Richards, *J. Mol. Biol.*, **55**, 379 (1977).
12. M. E. Davis and J. A. McCammon, *J. Comput. Chem.*, **12**, 909 (1991).
13. E. A. Padlan, E. W. Silverton, S. Sheriff, G. H. Cohen, S. J. Smith−Gill, and D. R. Davies, *Proc. Natl. Acad. Sci. USA*, **86**, 4592 (1989).
14. M. Born, *Zeit. Phys.*, **1**, 45 (1920).
15. J. A. Tainer, E. D. Getzoff, K. M. Beem, J. S. Richardson, and D. C. Richardson, *J. Mol. Biol.*, **160**, 181 (1982).
16. I. Klapper, R. Hagstrom, R. Fine, K. Sharp, and B. Honig, *Proteins: Struct. Funct. Genet.*, **1**, 47 (1986).
17. W. L. Jorgensen and J. Tirado−Rives, *J. Am. Chem. Soc.*, **110**, 1657 (1988).
18. J. N. Scarsdale, C. V. Alsenoy, V. J. Islimkowski, L. Schäfer, and F. A. Momany, *J. Am. Chem. Soc.*, **105**, 3438 (1983).
19. S. J. Weiner, P. A. Kollman, D. T. Nguyen, and D. A. Case, *J. Comput. Chem.*, **7**, 230 (1986).